ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет компьютерных наук
Образовательная программа бакалавриата «Программная инженерия»

СОГЛАСОВАНО
Научный руководитель,
профессор департамента
программной инженерии
факультета компьютерных наук
канд. техн. наук

_____ С.М. Авдошин
«___» _____ 2020 г.

УТВЕРЖДАЮ
Академический руководитель
образовательной программы
«Программная инженерия»
профессор департамента программной
инженерии, канд. техн. наук

_____ В.В. Шилов
«___» _____ 2020 г.

Подп. и дата

Инв. № дубл.

Взам. инв. №

Подп. и дата

Инв. № подл

ПРОГРАММА КЛАССИФИКАЦИИ КОМПЬЮТЕРНЫХ АТАК ПО НАБОРУ ДАННЫХ ISCX BOTNET

Программа и методика испытаний

ЛИСТ УТВЕРЖДЕНИЯ

RU.17701729.02.13-01 51 01-1-ЛУ

Исполнитель
студент группы БПИ193
_____ /Е.А. Гриценко /
«____»_____ 2020 г.

Москва 2020

УТВЕРЖДЕН
RU.17701729.02.13 -01 51 01-1-ЛУ

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл	

ПРОГРАММА КЛАССИФИКАЦИИ КОМПЬЮТЕРНЫХ АТАК ПО НАБОРУ ДАННЫХ ISCX BOTNET

Программа и методика испытаний

RU.17701729.02.13-01 51 01-1

Листов 14

Москва 2020

ОГЛАВЛЕНИЕ

1. ОБЪЕКТ ИСПЫТАНИЙ

1.1. Наименование ПО

ПО состоит из двух чатей: библиотеки классов и програмы. Наименование программы — “BotnetDetector”, наименование библиотеки — “LibBtntDtct”. Общее наименование ПО — “BotnetDetector”.

1.2. Область применения

Областью применения программы “BotnetDetector” является информационная безопасность. Под botnet-ом понимается сеть компьютерных устройств, в частности — IoT-устройств, заражённых вирусом (bot-ом), который предоставляет атакующему возможность удалённого массового выполнения на них команд и удалённого использования: в частности — для проведения DDOS атак, спам-рассылок, mining-а криптовалют.

Программа призвана помочь обеспечить защиту от botnet-ов, путём их обнаружения в сетевом трафике.

1.3. Наименование темы разработки

Наименование темы разработки — «Программа классификации компьютерных атак по набору данных ISCX Botnet» (“ISCX Botnet Dataset Classification Attack Program”).

2. ЦЕЛЬ ИСПЫТАНИЙ

Целью проведения испытания является проверка разработанного ПО на соответствие требованиям, изложенным в техническом задании, а именно — функциональным требованиям и проверка на отсутствие ошибок при соблюдённых условиях эксплуатации, валидности способа запуска и установки.

3. ТРЕБОВАНИЯ К ПРОГРАММЕ

При проведении испытаний функциональные характеристики программы подлежат проверке на соответствие функциональным требованиям, изложенным в разделе 4.1 технического задания. Также подлежит проверке работоспособность всех опций, предоставляемых программой.

4. ТРЕБОВАНИЯ К ПРОГРАМНОЙ ДОКУМЕНТАЦИИ

Состав программной документации должен соответствовать указанному в техническом задании.

4.1. Состав программной документации

- «Программа классификации компьютерных атак по набору данных ISCX Botnet». Техническое задание;
- «Программа классификации компьютерных атак по набору данных ISCX Botnet». Текст программы;
- «Программа классификации компьютерных атак по набору данных ISCX Botnet». Пояснительная записка;
- «Программа классификации компьютерных атак по набору данных ISCX Botnet». Руководство оператора;
- «Программа классификации компьютерных атак по набору данных ISCX Botnet». Руководство программиста.

5. СРЕДСТВА И ПОРЯДОК ПОРЯДОК ИСПЫТАНИЯЙ

5.1. Технические средства.

Испытание программы производится на ЭВМ “HP EliteBook 8570p A1029D1102”, оснащённый 2-х ядерным 64-битным процессором Intel i7-3520M поколения IvyBridge, имеющий 16 GB RAM.

5.2. Программные средства.

Иыспытнаия программы производится на операционных системах:

- Arch Linux, версия сборки mono 5.18.0.240-1, libpcap 1.9.1-2, остальные пакеты являются актуальными стабильными релизами на 05.05.2020 20:14 (UTC+3).

- Microsoft Windows 10.

На каждой из ОС используется надлежащим образом установленные файлы разработанного ПО. Способы установки приводятся в руководстве оператора.

5.3. Порядок испытаний.

Соответствие библиотеки классов каждому пункту требований к её функциональным характеристикам тестируется с помощью программы её использующей. Соответствие программы каждому пункту требований к функциональным характеристикам программы проверяется на основании тестов, указанных в методике испытаний. Проверка должна производится на обеих системах.

Помимо этого должен быть, согласно специальным требованиям в главе 4.8 ТЗ, проверена точность и работоспособность классификатора программы на основе тестовой части набора данных “ISCX Botnet 2014”.

В дополнение к приведённым, допускается проверка точности классификатора программы на иных наборах данных и образцах трафика, захваченного с указанной в технических средствах ЭВМ и иных ЭВМ.

Работоспособность показа статистики о классифицированных потоках, опции произведения классификации, опции выбора пользовательского классификатора, опции печати (печатать информацию о потоке, о его классе, или же и о потоке, и о классе), опции использования бинарной классификации на основе текущего классификатора, опции печати информации только для тех потоков, класс которых не является нормальным, инвариантны относительно источника информации о потоках и способа вывода (в файл или в стандартный вывод), в силу того, что этот функционал прописан для базового

класса для реализующих пункты 4.1.1.1 — 4.1.1.3 (включительно) ТЗ. Поэтому из работоспособности одной из перечисленных опций для какого-либо из источников и способов вывода следует работоспособность этой опции для всех источников и способов вывода.

Перед началом испытаний необходимо корректно установить приложение, обеспечив загрузку всех его dll-зависимостей и обеспечить иные условия выполнения програмы. О возможных путях написано в руководстве оператора. Также необходимо удостовериться, что у программы и пользователя есть разрешения на прослушивание трафика, чтение и запись файлов.

5.4 Методика испытаний.

Пусть корректная комманда запуска “botnetdetector”. Пусть “testing.csv” и “training.csv” — пути к файлам с маркированными потоками тестовой и тренировочной части набора данных “ISCX Botnet 2014” (информацию об этих файлах и о том, как они были получены можно найти в главе 3.6 пояснительной записки), “testing.pcap” и “training.pcap” — к файлам дампов трафика вышеописанных частей. Пусть “eth” — название устройства с активным Ethernet/IP/TCP трафиком, пакеты которого корректно парсируются SharpPcap. Пусть “NonBinary.dll” — относительный путь к небинарному классификатору, отличающийся от абсолютного. Пусть “absolute/NonBinary.dll” — абсолютный путь к этому классификатору. Пути при наличии символов, контролирующих поведение терминала или поведение его парсера аргументов должны записываться способом, позволяющим избежать этих изменений (например, при наличии пробелов, таким способом может быть обособление его кавычками).

Необходимо

5.4.1. Выполнить “botnetdetector”. Проверка запускаемости. Ожидаемый результат: сообщение-помощь с перечислением опций программы. Наличие ошибок инициализаций типа свидетельствует о неверной установке или способе запуска. Их отсутствие удостоверяет запускаемость приложения и корректную связь с .NET-библиотеками.

5.4.2. Выполнить “botnetdetector -ls” и “botnetdetector --listDevicies”. Ожидаемый результат: список доступных сетевых устройств, если они есть. Наличие ошибок может свидетельствовать об отсутствии прав у пользователя на прослушивание сетевых устройств, отсутствие зависимостей для SharpPcap (libpcap для Linux, npcap для Windows),

иные варианты некорректной установки, либо же ошибку непосредственно в программе. Наличие ожидаемого результата означает корректную связь приложения с библиотеками для работы с сетевым трафиком.

5.4.3. Выполнить “botnetdetector -ls -h”. Ожидаемый результат: сообщение-помощь с перечислением опций программы. Отсутствие этого результата означает ошибку в обработке программой аргументов.

5.4.4. Выполнить “botnetdetector -rand_string”, где “-rand_string” любая ненулевая строка из латинских символов с возможными включениями символа ‘-’, не равная ни одной из строк опций. Ожидаемый результат: строка “Unrecognised option -rand_string”. Отсутствие этого результата означает ошибку в обработке программой аргументов.

5.4.5. Выполнить “botnetdetector -o”. Ожидаемый результат: сообщение о том, что самая правая опция требует больше аргументов. Отсутствие этого результата означает ошибку в обработке программой аргументов.

5.4.6. Выполнить “botnetdetector -i testing.pcap -p --output test.csv”. Ожидаемый результат: во время выполнения — отображаемая посекундно статистика о количестве прочитанных пакетов и количестве классифицированных потоков: как нормальных и как ботнет-потоков, CSV-файл test.csv с информацией о классифицированых потоках формата, описанного в пункте 4.1.2.4 ТЗ.

5.4.7. Выполнить “botnetdetector -o test.csv --input testing.pcap --pcap --classify -0”. Ожидаемый результат: отсутствие статистики о классифицированных потоках и информации о классах в test.csv, test.csv соответствует формату, описанному в пункте 4.1.2.3. ТЗ.

5.4.8. Выполнить “botnetdetector --printFlows -0 --input testing.pcap --pcap”. Ожидаемый результат: печать в терминале информации о классах выявленных в файле потоков, соответствующей формату, описанному в пункте 4.1.2.5 ТЗ.

5.4.9. Выполнить “botnetdetector -i eth -o test_noheader.csv --printClasses -0 -l”. Через две минуты с половиной нажать сочетание клавиш, отменяющее команду (“Ctrl+C” для Linux). Ожидаемый результат: статистика о прочитанных пакетах и классифицированных потоках, обновляемая каждую секунду; по завершении программы test_noheader.csv должен содержать информацию о потоках без информации о классах и соответствовать формату, описанному в пункте 4.1.2.3 ТЗ.

5.4.10. Выполнить “botnetdetector --printAbnormalOnly --live -i eth”. Через две минуты нажать сочетание клавиш, отменяющее команду (“Ctrl+C” для Linux). Ожидаемый результат: в терминале должна быть напечата информациях только для потоков, не принадлежащих классу нормальных.

5.4.11. Выполнить “botnetdetector --table -i testing.csv”. Ожидаемый результат: строка “error reading flow #0”.

5.4.12. Выполнить “botnetdetector -t -i testing_noheader.csv”. Ожидаемый результат: печать в терминале информации о потоках и классах.

5.4.13. Выполнить “botnetdetector --table -i testing.csv --skipFirstLine -o test.csv”. Ожидаемый результат: во время выполнения — отображаемая посекундно статистика о количестве прочитанных потоков и количестве классифицированных потоков: как нормальных и как ботнет-потоков, CSV-файл test.csv с информацией о классифицированых потоках.

5.4.14. Выполнить “botnetdetector --classifier NonBinary.dll -i testing.csv”. Ожидаемый результат: строка “Unrecognised option testing.csv”.

5.4.15. Выполнить “botnetdetector --classifier NonBinary.dll NonBinary.Tree -i testing.csv”. Ожидаемый результат: сообщение об ошибке загрузки классификатора.

5.4.16. Выполнить “botnetdetector -i testing.pcap --classifier NonBinary.dll NonBinary.Classifier --printFlows -0 -p”. Ожидаемый результат: печать в терминале строк с именами классов потоков, не содержащая класс “botnet” встроенного классификатора.

5.4.17. Выполнить “botnetdetector --nameAbnormalAsBotnet -i testing.pcap --classifier NonBinary.dll NonBinary.Classifier --printFlows -0 -p”. Ожидаемый результат: печать в терминале строк, каждая из которых либо “botnet”, либо “normal”.

5.4.18. Выполнить “botnetdetector -i testing.pcap --printFlows -0 --printClasses -0 --pcap”. Ожидаемый результат: во время выполнения — отображаемая посекундно статистика о количестве прочитаных пакетов и количестве классифицированных потоков: как нормальных и как ботнет-потоков.

5.4.19. Выполнить “botnetdetector --classifier NonBinary.dll NonBinary.Classifier -pr -i training.csv --skipFirstLine”. Ожидаемый результат по окончании выполнения: статистика по прочитанным и вычисленным классам и точности классификатора по файлу training.csv.

5.4.20. Выполнить “botnetdetector --performance -i training.csv --skipFirstLine”. Ожидаемый результат по окончании выполнения: статистика по прочитанным и вычисленным классам и точности встроенного классификатора по файлу testing.csv. Таким образом, выполняется специальное требование о тестировании на основе тестовой части набора данных “ISCX Botnet 2014” главы 4.8. ТЗ.

5.4.21. Выполнить “botnetdetector -i testing.pcap -o test.csv -v -t” и “botnetdetector -t -i testing.pcap -o test.csv --verbose”. Ожидаемый результат: сообщение об ошибке с полным следом стэка.

5.4.22. Выполнить “botnetdetector -i testing.pcap -p -q -o test.csv” и “botnetdetector -i testing.pcap --quiet -p -o test.csv”. Ожидаемый результат: CSV-файл test.csv с информацией о классифицированых потоках; при этом во время выполнения не дожно происходить печати статистики.

5.4.23. Выполнить “botnetdetector -i testing.csv --printFlows -0 --printClasses -0 --classifier Alpha Beta --printClasses -1 --classifier -0 --skipFirstLine -1 -t”. Ожидаемый результат: печать в терминале информации о классах потоков.

5.4.24. Выполнить “botnetdetector -i testing.csv --absoluteClassifierPath --classifier NonBinary.dll NonBinary.Classifier -t”. Ожидаемый результат: сообщение об ошибке загрузки классификатора.

5.4.25. Выполнить “botnetdetector --classifier absolute/NonBinary.dll NonBinary.Classifier --absoluteClassifierPath -pr -i training.csv --skipFirstLine”. Ожидаемый результат по окончании выполнения: статистика по прочитанным и вычисленным классам и точности классификатора по файлу training.csv.

Отсутствие ожидаемого результата для хотя бы одного их вышеописанных тестов свидетельствует о наличии ошибок в программе или об ошибках, допущенных в процессе проведения тестирования. Выполнение тестов 5.4.6 и 5.4.7, тестов 5.4.9 и 5.4.10, тестов 5.4.11 и 5.4.12, тестов 5.4.19 и 5.4.20, тестов с 5.4.6-ого по 5.4.20-ый включительно, означают выполнение подпунктов 4.1.1.2.1, 4.1.1.2.2, 4.1.1.2.3, 4.1.1.2.5 и 4.1.1.2.4 пункта ТЗ. Выполнение всех этих подпунктов, в силу того, что приложение основано на библиотеке, определяемой требованиями пункта 4.1.1.2, означает выполнение всех подпунктов этого пункта. Таким образом, если ожидаемый результат наличествует на

тестах с 5.4.6-ого по 5.4.25-ый, ПО отвечает всем требованиям функциональных характеристик.

6. СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. «Программа классификации компьютерных атак по набору данных ISCX Botnet». Техническое задание;

2. «Программа классификации компьютерных атак по набору данных ISCX Botnet». Текст программы;

3. «Программа классификации компьютерных атак по набору данных ISCX Botnet». Пояснительная записка;

4. «Программа классификации компьютерных атак по набору данных ISCX Botnet». Руководство оператора;

5. «Программа классификации компьютерных атак по набору данных ISCX Botnet». Руководство программиста.

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

Изм.	Номера листов (страниц)				Всего листов (страниц) в документе	№ документа	Входящий № сопроводительного документа и дата	Подпись	Дата
	измененных	замененных	новых	аннулированных					